

**AMENDMENTS TO THE CLAIMS**

This listing of claims will replace all prior versions, and listings, of claims in the application:

1 to 45. (cancelled)

46. (new) A method of journaling changes to system objects including the steps of:

- i) substituting a dummy function for a system function;
- ii) executing the system function under operation of the dummy function;
- iii) generating copies of system objects, changed by execution of the system function, for journaling; and
- iv) completing execution of the dummy function.

47. (new) A method as claimed in claim 46 wherein the dummy function substitutes the system function by having a duplicate calling name and pre-empting the execution of the system function.

48. (new) A method as claimed in claim 47 wherein an exit point is associated with the dummy function and an exit program registered for the exit point.

49. (new) A method as claimed in claim 48 wherein during operation of the dummy function the exit program is executed.

50. (new) A method as claimed in claim 49 wherein the execution of the system function is handled by the exit program.
51. (new) A method as claimed in claim 50 wherein the system objects changed by execution of the system function are captured by the exit program.
52. (new) A method as claimed in claim 51 wherein the exit program generates copies of the system objects captured by the exit program.
53. (new) A method as claimed in claim 49 wherein the execution of the system function is handled by the dummy function.
54. (new) A method as claimed in claim 53 wherein the system objects changed by execution of the system function are captured by the dummy function.
55. (new) A method as claimed in claim 54 wherein the exit program generates copies of the system objects captured by the dummy function.
56. (new) A method as claimed in claim 52 wherein the copies of the system objects are saved to disk.

57. (new) A method as claimed in claim 52 wherein the copies of the system objects are streamed to a database system for journaling.

58. (new) A method as claimed in claim 57 wherein the database system is incorporated with a replication system.

59. (new) A method as claimed in claim 58 wherein the replication system replicates the copies of the system objects to one or more local or remote databases.

60. (new) A method as claimed in claim 52 wherein messages or exceptions generated by the system function are captured into a queue.

61. (new) A method as claimed in claim 60 wherein the system function is originally called by a process and the messages or exceptions are forwarded back to the process by the dummy function.

62. (new) A method as claimed in claim 52 wherein the system objects are one or more of the set of program objects, configuration objects, queues, and space/memory mapped objects.

63. (new) A method as claimed in claim 52 wherein the changed system objects are those system objects which have been created, changed or deleted.

64. (new) A method as claimed in claim 52 wherein the system functions are OS/400 system functions.

65. (new) A method of journaling changes to system objects including the steps of:

- v) executing a system function during which changes to system objects occur; and
- vi) journaling changes to system objects during execution of the system function.

66. (new) A method as claimed in claim 65 wherein changes to system objects are journaled by integrating journaling commands into the code of the system functions.

67. (new) A method as claimed in claim 65 wherein changes to system objects are journaled by associating exit points with the system function and calling an exit program during execution of the system function.

68. (new) A system for journaling changes to system objects including:

- vii) a processor adapted to execute a dummy function in place of a system function wherein the dummy function executes the system function and generates copies of system objects resulting from system function execution for journaling; and
- viii) memory for use by the processor during execution.

69. (new) A system as claimed in claim 68 wherein the dummy function substitutes the system function by having a duplicate calling name and pre-empting the execution of the system function.

70. (new) A system as claimed in claim 69 wherein an exit point is associated with the dummy function and an exit program registered for the exit point.

71. (new) A system as claimed in claim 70 wherein during execution of the dummy function the exit program is executed.

72. (new) A system as claimed in claim 71 wherein the execution of the system function is handled by the exit program.

73. (new) A system as claimed in claim 72 wherein the system objects changed by execution of the system function are captured by the exit program.

74. (new) A system as claimed in claim 72 wherein the exit program generates copies of the system objects captured by the exit program.

75. (new) A system as claimed in claim 71 wherein the execution of the system function is handled by the dummy function.

76. (new) A system as claimed in claim 75 wherein the system objects changed by execution of the system function are captured by the dummy function.

77. (new) A system as claimed in claim 76 wherein the exit program generates copies of the system objects captured by the dummy function.

78. (new) A system as claimed in claim 74 wherein the copies of the system objects are saved to disk.

79. (new) A system as claimed in claim 74 wherein the copies of the system objects are streamed to a database system for journaling.

80. (new) A system as claimed in claim 79 wherein the database system is incorporated with a replication system.

81. (new) A system as claimed in claim 80 wherein the replication system replicates the copies of the system objects to one or more local or remote databases.
82. (new) A system as claimed in claim 74 wherein messages or exceptions generated by the system function are captured into a queue.
83. (new) A system as claimed in claim 82 wherein the system function is originally called by a process and the messages or exceptions are forwarded back to the process by the dummy function.
84. (new) A system as claimed in claim 74 wherein the system objects are one or more of the set of program objects, configuration objects, queues, and space/memory mapped objects.
85. (new) A system as claimed in claim 74 wherein the changed system objects are those system objects which have been created, changed or deleted.
86. (new) A system as claimed in claim 74 wherein the processor is an AS/400 processor.
87. (new) A system as claimed in claim 86 wherein the processor is operating under the OS/400 operating system.

88. (new) A computer system for effecting the method of claim 46.
89. (new) A computer system for effecting the method of claim 65.
90. (new) Software for effecting the method of claim 46.
91. (new) Software for effecting the method of claim 65.
92. (new) Storage media containing software as claimed in claim 90.
93. (new) Storage media containing software as claimed in claim 91.